

# Review of Macromedia's RoboHelp X5

by Adele Sommers

San Luis Obispo Chapter of the Society for Technical Communication (STC)

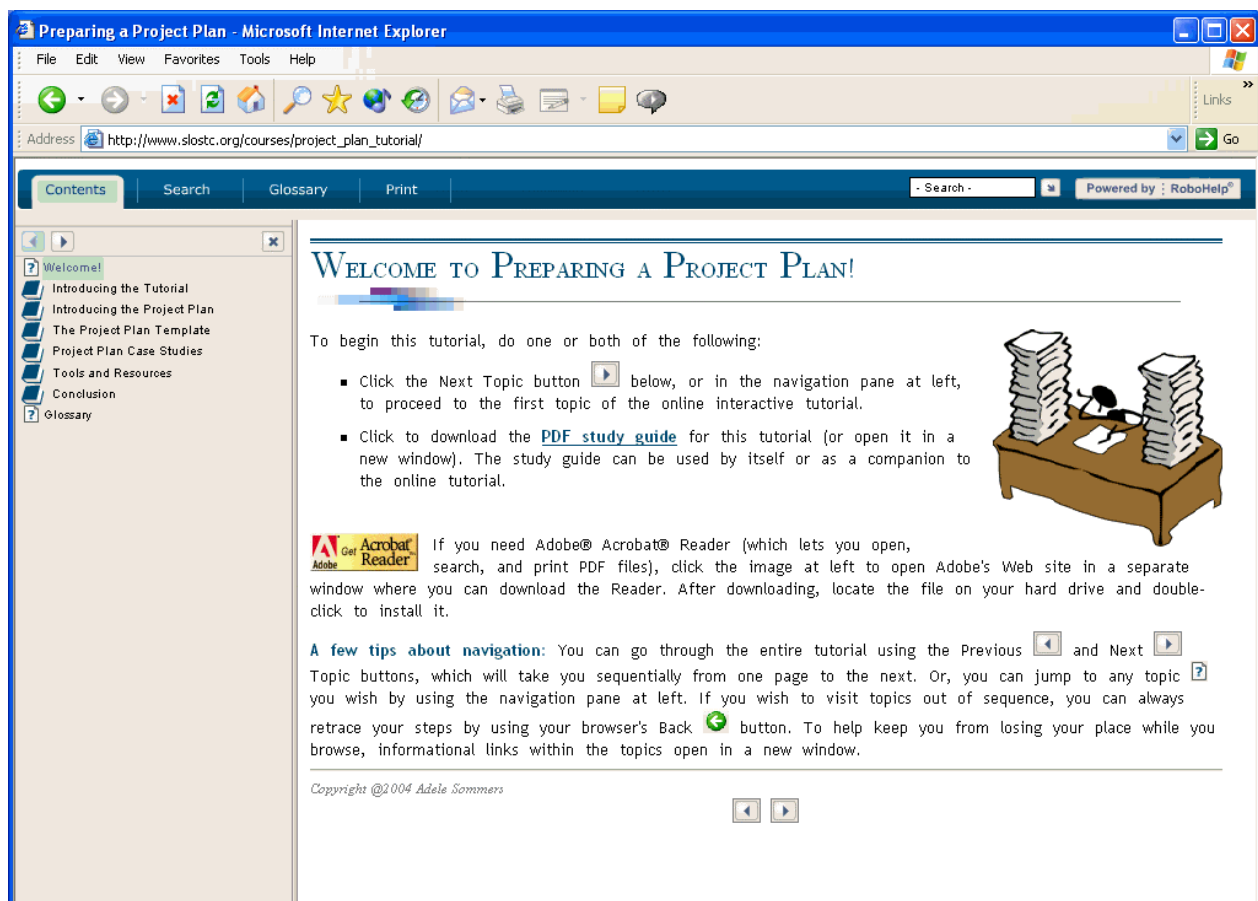
<http://slostc.org>

## Help Authoring Systems Are Not Just for Creating Help

To evaluate RoboHelp X5, I selected a project involving a tutorial rather than a typical Help system. I'm always looking for ways to make one tool serve multiple purposes (perhaps another aspect of "single-sourcing"), so I was anxious to determine whether X5 would work as well for tutorials and marketing presentations as I assumed it would for Help systems. Since I've previously been able to undertake these types of projects with Help authoring applications, I wanted to try at least one example for this review.

With any project, I prefer to have strong control over the look, feel, and navigability of the final product. For this endeavor, the result also needed to perform well across platforms and browsers, such as Macintosh OS 9, OS X, and Windows XP.

The tutorial I ultimately completed is called "Preparing a Project Plan," which is now part of the Best Practices Library of the San Luis Obispo STC. The tutorial's launching page resides at [http://www.slostc.org/topics/project\\_mgmt/overview.html#tutorial](http://www.slostc.org/topics/project_mgmt/overview.html#tutorial). Once a visitor launches the tutorial, its splash page opens as shown below.



## Why I Selected a Tutorial Project

One reason for choosing an instructional project was to evaluate the range and flexibility of the navigation options that I could embed in the presentation. A tutorial with both sequential and discretionary navigation options (such as the "Previous" and "Next" buttons of a browse sequence and the navigation pane, respectively) can help learners achieve a far more satisfying experience than one that offers a single option. Similarly, where marketing presentations are concerned, providing multiple navigation alternatives, as well as the ability to hide or show the navigation pane, can be equally beneficial.

In selecting a tutorial, I also wanted to demonstrate the ability to create a robust cross-platform-and-browser framework for including any sort of HTML or JavaScript-based content without incurring the expense of standard Web-Based Training (WBT) development. From past experience, I knew that a Help system with the right set of customizable bells and whistles can help an instructional designer produce a very functional learning system at a fairly low cost and with few relatively ingredients.

The low-cost development scenario increases in importance as companies continue to downsize both technical communication and training departments. In situations where fewer specialists remain to cover traditional content design and development bases, the challenge often rests with generalists to perform specialized tasks with limited resources. This trend underscores the need to combine flexible authoring tools with diverse examples and tutorials. Such resources can spark the authors' imaginations and demonstrate what they can accomplish in different contexts.

## How I Created the Tutorial

My goal for the project was to produce an interactive, online tutorial and companion PDF self-study guide from a single set of FrameMaker source files. I designed the source files using conditional tags that would generate differentiated content for the interactive tutorial and the printable companion guide.

I separately created the following components to become part of the tutorial system:

- ◆ The PDF guide, which I generated from FrameMaker using Adobe Acrobat Distiller.
- ◆ The JavaScript components (the assessments and exercises), for which I used Dreamweaver and its free CourseBuilder extension.

In an initial attempt to construct the interactive tutorial from my FrameMaker source files, I tried using Macromedia's new RoboHelp for FrameMaker software, which I had recently purchased. As promised by its advertising, this tool does indeed wrap itself seamlessly around FrameMaker files. Although I believe it holds enormous promise for FrameMaker enthusiasts like me, I found that it is still very much in its infancy. To my dismay, I soon discovered that certain basic things I needed to do, such as create browse sequences, were not yet supported in version 1.1.728.6.

Fortunately, I was able to switch applications and adapt fairly quickly to RoboHelp X5. Although it can't edit FrameMaker source files directly, X5 can import FrameMaker's interchange format (.mif) files. I decided to bank on X5 for completing the tutorial.

Past experience with other Help systems aided me conceptually because I generally knew what to expect. These included a much earlier version of RoboHelp I had once tried, over which the user interface for the X5 version was an amazing improvement. Other than that fairly early RoboHelp experience, I had never used RoboHelp Office.

The tutorial that came with X5 effectively demonstrated its primary functions. After reviewing the tutorial, I was able to use the New Project Wizard to set up my project and import the .mif files. I was quite pleased with the control I could exert over the styles and pagination during the importing process. The automatic creation of topics based on FrameMaker styles worked well, setting the stage for applying templates to and fine-tuning various topics. I was also able to perform other housekeeping tasks without much difficulty, including creating a Table of Contents and browse sequences.

By the time I was finished with my tutorial project, I had pre-tested, created, tested, published, and post-tested the tutorial on the designated Web site. Further details of my experiences appear in the remaining sections of this review.

## A Summary of My Positive Results

There are many features to crow about with this impressive tool. Here are highlights of the many things I came to appreciate about the X5 interface and its functionality:

- ◆ **Wizards** – Wizards exist not only for setup processes but also for certain feature-rich Properties dialog boxes. For the latter purpose, the wizards help ensure that the author does not overlook key settings that might otherwise become buried.
- ◆ **Styles** – An author can handily manage character and paragraph styles through one or more Cascading Style Sheets (CSS). The dialog box for creating or updating styles looks and functions very much like that of Microsoft Word.
- ◆ **Shortcuts** – Right-mouse menus provide context-specific shortcuts to just about every desirable feature in the system. For example, right-clicking in the body of a displayed topic enables the author to quickly access font styles, paragraph styles, and other formatting features; insert or edit images; insert or edit hyperlinks, manage topic properties, and preview the topic.
- ◆ **Glossary entries** – X5 provides several ways of presenting definitions of glossary terms, including popup windows and expandable text for designated terms in the body of the Help topics. Because some methods require target browsers to support DHTML functionality, the author should first test them in the appropriate context.
- ◆ **Navigation pane** – Specifically, how the navigation pane loads and displays in the browser is a key issue where cross-platform-and-browser output is concerned. A different Help authoring system I had used in the past did not offer control over the navigation pane's formatting, particularly where Java applets were used to generate the display of the navigation pane. (Although Java applets work across platforms and browsers, I have found that they tend to load slowly, require the Java Runtime Environment plug-in, and may not work from inside a firewall.) In contrast, RoboHelp X5 offers flexible choices for building the navigation pane for WebHelp output (RoboHelp's HTML-based cross-platform-and-browser format). The author can choose to include or exclude both Java applets and Dynamic HTML (DHTML) components in addition to using pure HTML.

- ◆ **Single source layouts** – An author can quickly switch between the single source layout types (such as WebHelp, FlashHelp, Microsoft HTML Help, and XML) without doing more than choosing or changing a few settings.

Although this capability has long been a standard with Help authoring systems, I was impressed with the clean interface to the available options and their settings. These features enabled me to carry out cross-platform-and-browser tests to select the optimum format for my needs. After a careful comparison of the performance of WebHelp and FlashHelp in my target platforms and browsers, I finally selected FlashHelp because of its more consistent behavior across browser environments.

I should note that the FlashHelp format uses Flash mainly to build the navigation and tool bar portions of the user interface. The main information pane remains HTML-based, which means that cross-browser-and-platform limitations related to HTML and DHTML still apply to this pane.

- ◆ **Popups** – An author can set an option to have a link to another topic open in a custom-sized popup window, if desired, rather than the main information pane. This feature is critical whenever it's important to avoid forcing the user out of a particular flow merely to view optional information. In previous Help authoring experiences, I had found that popups tended not to work well outside of Internet Explorer 5.x and later for Windows, for example. To my surprise and delight, I discovered that the X5 popups opened quickly and gracefully in IE 5.x and later and Netscape 7.x for both Windows and Macintosh, as well as in a few other browsers I tested.
- ◆ **WYSIWYG and TrueCode editors** – The ability to toggle back and forth between a WYSIWYG editor and a coding/scripting environment is an important element of a multipurpose authoring system. Although I mainly used the WYSIWYG editor, I appreciated the clean and functional interfaces to both editing environments.
- ◆ **Customizable skins** – Perhaps one of the most welcome aspects of the system is the growing collection of skins, or interface styles, available for the various single source layouts. An author can customize a skin by changing features such as the color scheme, fonts, and buttons. (Unfortunately, this process is not as easy with FlashHelp as it is with WebHelp because it requires creating or editing components in Flash. I understand, however, that a future enhancement will make FlashHelp skins much easier to modify.) The skins come packaged with the system and also can be downloaded from Macromedia's eHelp support Web site.
- ◆ **Previewing** – Authors can preview their work in various ways without having to publish the entire project. The handiest methods include the right-mouse options associated with individual topics and with any of the single source layouts. In the preview mode, topics look and behave just as they would in a published project.
- ◆ **Build options** – When it comes time to generate and publish the entire project (or portions thereof, by using conditional tags in the topic properties), the Properties dialog box for any of the single source layouts makes this task a breeze. Publishing to either a local hard drive destination or to one or more Web sites works very well. The FTP settings are straightforward, and one can specify and store as many Web site targets as desired.

Please note that my many other positive X5 experiences are too numerous to list here.

## Key Features I Didn't Test

Although RoboHelp X5 offers a wealth of very useful and powerful capabilities, there were many I did not have the opportunity to try. I relegated for testing on future projects features such as:

- ◆ RoboSource Control content management features
- ◆ The use of Dreamweaver or Word as a primary authoring tool
- ◆ Distributed workforce support features
- ◆ Multi-author support features
- ◆ Microsoft HTML Help output
- ◆ Context-sensitive Help
- ◆ XML support features
- ◆ PDF publishing

## Suggestions for Enhancements

The section discusses a list of features that presently seem to be missing from the system, but would add significant benefit to Help authors if addressed.

### **1) The system needs a self-invoking utility that automatically inspects and repairs a corrupted database.**

I experienced a major setback during my trial in which the main project database became corrupt after a full day of work. I discovered this when I attempted to reopen the project and received an error about the corruption. Contrary to my expectations, no integrated utility automatically examined and attempted to repair the database — a standard feature in some other Help systems and many applications I have used.

I requested help via e-mail. Macromedia's Technical Support responded in a timely manner, but suggested a complex process: "You can try renaming both the .cpd and the .xpj files, and then opening up the .hhp file [the critical information container]. This will rebuild both the .xpj and the .cpd file from information that it has stored. Usually this will have all of the topic information stored, so you should be able get a project from it. If that doesn't work, import the .htm files back into the project. All the information should still be there and the links should be recreated."

Unfortunately, these actions did not restore my project. Portions of the system were recreated, but were also missing links, data, and other connections. Ultimately, I used a nightly backup of the previous day's work and repeated all of my lost transactions.

Technical Support also suggested that the X5 RoboSource Control utility constitutes the equivalent of a recovery tool because you can roll back to ("restore") a particular system state. But the purpose of a content management system such as RoboSource Control is to store snapshots of the system, not necessarily to perform an integrity check on the database. I was referring to a *repair* utility that automatically looks for corruption or damage in the current version of the database and fixes it.

Upon inquiring further about whether a repair utility might become a future system enhancement, I received a very welcome phone call from Macromedia's Technical

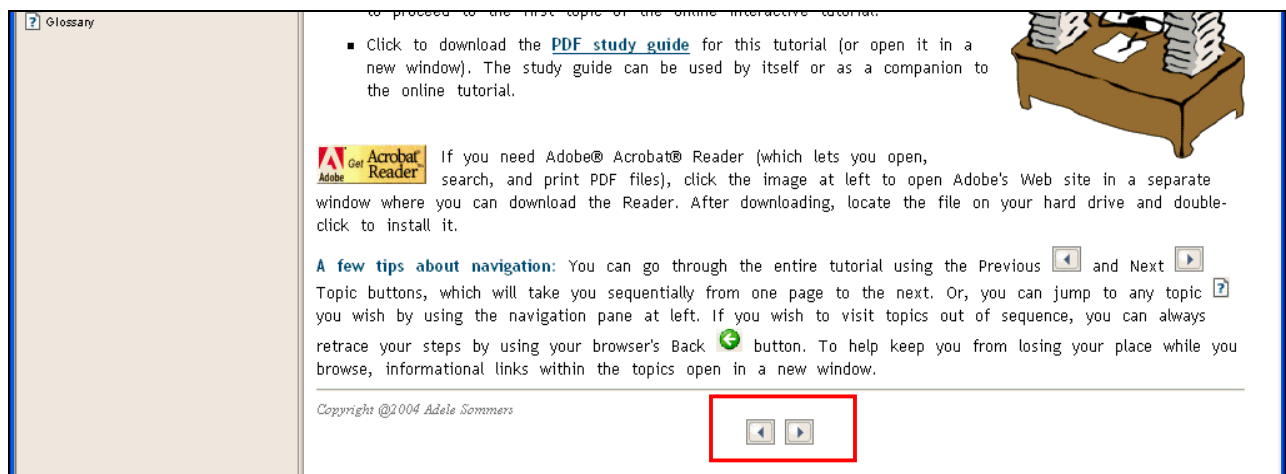
Support Lead, Raul Ramos. Raul listened at length to my concerns and was quick to acknowledge the need for such a tool. He also noted some of my other suggestions for enhancements and further invited me to participate in a weekly Webinar he gives on X5 installation and configuration, which I subsequently attended.

## **2) The system should make it possible to generate browse sequence (“Next” and “Previous”) buttons automatically and place them anywhere in the main display, such as in the header or footer of the information pane, or both.**

Currently, the browse sequence buttons reside only in the upper left corner of the navigation pane. This placement creates a usability problem, since much of the user population is right-handed. It’s also a distraction, since users are focusing primarily on the main information window, not the navigation pane.

According to Raul Ramos, the ability to generate and place browse sequence buttons in other locations does exist, but only if you are generating WebHelp output with the “Classic” WebHelp skin. Incorporating this capability into other output types, such as FlashHelp, and with all types of skins, would make it much easier for an author to create the sequential navigation more typical of marketing presentations and tutorials. My workaround for this limitation was to create a set of Next and Previous buttons that matched the skin I was using, embed them into the Help page template’s header and footer, and manually link them to the correct pages in the browse sequence (a time-consuming process).

The cropped image below shows a red box around the buttons I created and centered in the page footer. I also replicated these buttons in the right side of the header of all pages except the splash page. The result gives three dispersed navigation points: The left-hand navigation pane, the bottom of each page, and upper right side of each page. Having three navigation points makes it possible for the user to either skim through the entire sequence without scrolling down, or if one chooses to scroll down, continue navigating from the bottom of the page without using extra arm, wrist, and hand movements.



## **3) The system should support custom window formats, sizes, and display characteristics for WebHelp and FlashHelp output without limitation.**

The ability to customize the look and feel of a display window in every respect should be a standard feature of the system, without having to use a secondary tool interface. I did find that an author can tailor the look of navigation panes and tool bars using skins, custom buttons, and hiding or showing certain functions, such as the Search or Index options. However, the author cannot customize window and pane sizing, specify the use of fewer than three panes, or modify captions or window placement unless he or she is also using the RoboHelp context-sensitive Help API. Since I didn't have access to this API, I was unable to customize my tutorial format as I would have liked.

#### **4) Miscellaneous enhancement suggestions:**

- a. It would be very desirable to easily:
  - Copy and modify a style.
  - Reset the size of an image to its original dimensions.
  - Remove automatically generated hyphens from imported .mif files. (If not removed, they tend to display in a browser as character anomalies, even though they are not visible in the WYSIWYG editor.)
- b. Have the Preview mode offer a choice of target browsers.
- c. When choosing a hyperlink option to display something in a new HTML window, have the ability to specify the window's dimensions and characteristics. (The ability does exist for customizing the size of popups, but it would be nice to also control the same characteristics for a new HTML window.)
- d. When constructing a Conditional Build Expression based on the Conditional Build Tags set within the Topic Properties, it would be ideal to be able to select or deselect topics associated with a particular tag using a "laundry list" method. Such a list would display all of the topics with checkboxes next to them, letting the author apply a tag to, or remove a tag from, either selected topics or all topics. This approach would avoid the need to go into each topic individually to change its settings.

## **Conclusion**

I was delighted to have the opportunity to review this excellent tool for a substantive instructional design project. RoboHelp X5 met the majority of my needs for tutorial authoring, and with the modest enhancements I've suggested, I believe it would be an exemplary system for this kind of project. The next item on my "wish list" would be to incorporate the same set of capabilities into RoboHelp for FrameMaker.

RoboHelp X5's ease of use and wide range of robust features proved themselves worthy of recommendation to other authors of Help systems, tutorials, and sales and marketing presentations. By further expanding the ability of authors to adapt the look and feel of a presentation (through skins and other customizable elements), Macromedia could provide a very welcome opportunity for those, like me, who can visualize innumerable ways to apply a Help authoring tool. And by including tutorials and samples of instructional projects and non-Help-related presentations, I believe Macromedia will entice a far larger audience to explore the RoboHelp product line.